

# Spring MVC

## Spring AU 2021

Ajay S

[ajay.s@accolitedigital.com](mailto:ajay.s@accolitedigital.com)

---

### Assignment:

Create a Spring MVC application [properly MVC architecture]  
Order management application

API -> create order

API -> add Item

API -> view the items

API -> delete item

API -> update item description (any field)

-> Dependency injection wherever applicable

-> Validations and error handling wherever applicable

Code has been attached as a separate folder

### Endpoints:

1. Create order

<http://localhost:8080/api/order/create>

### Sample Data:

```
{
  "orderId" :1,
  "itemList": [{
    "itemId" : 1,
    "itemName": "Demo",
    "itemDescription": "Demo desc",
    "itemQuantity": 10
  },
  {
    "itemId" : 2,
```

```
    "itemName": "Demo 2",  
    "itemDescription": "Demo desc ",  
    "itemQuantity": 1  
  ]  
}
```

## 2. Add Item

<http://localhost:8080/api/order/addItem/{orderId}>

### Sample Data

```
{  
  "itemId" : 3,  
  "itemName": "Demo 3",  
  "itemDescription": "Demo desc",  
  "itemQuantity": 10  
}
```

## 3. View Items

<http://localhost:8080/api/order/getOrder/{orderId}>

### Sample response data

```
{  
  "orderId": 1,  
  "itemList": [  
    {  
      "itemId": 1,  
      "itemName": "Demo",  
      "itemDescription": "Demo desc",  
      "itemQuantity": 10  
    },  
    {  
      "itemId": 2,  
      "itemName": "Demo 2",  
      "itemDescription": "Demo desc ",  
      "itemQuantity": 1  
    }  
  ]  
}
```

#### 4. Delete Item

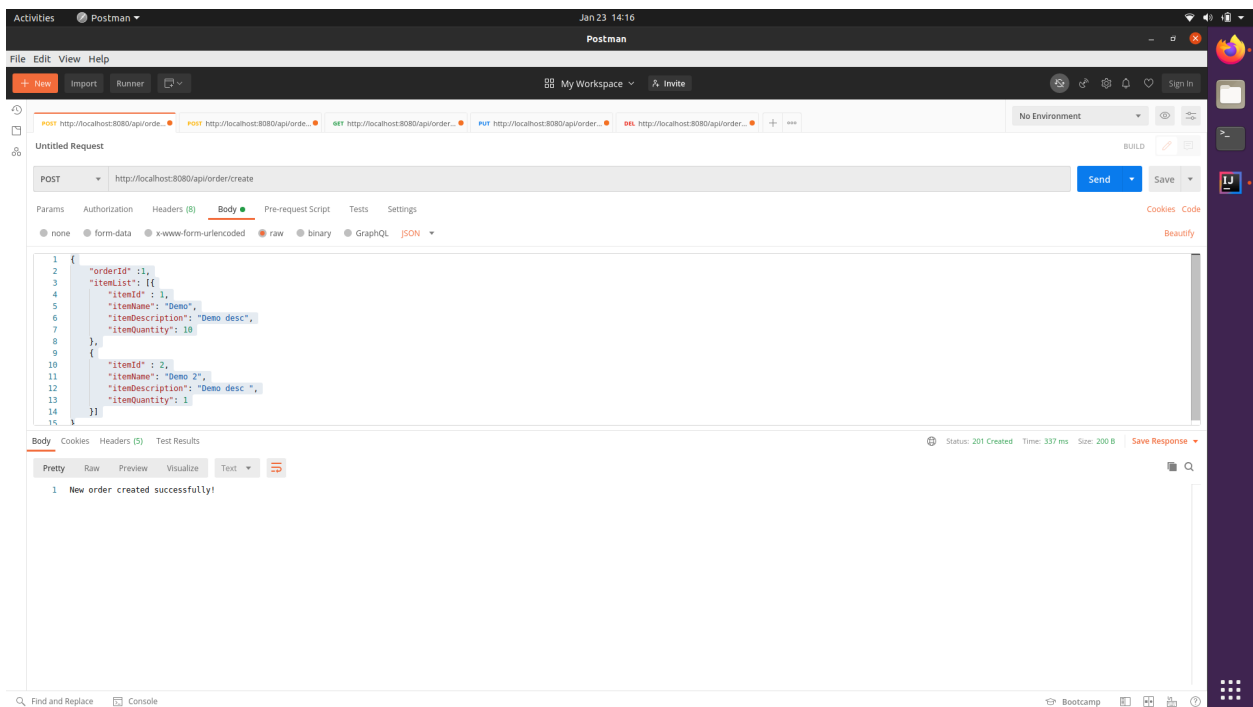
<http://localhost:8080/api/order/deleteItem/{orderId}/{itemId}>

#### 5. Update Item quantity

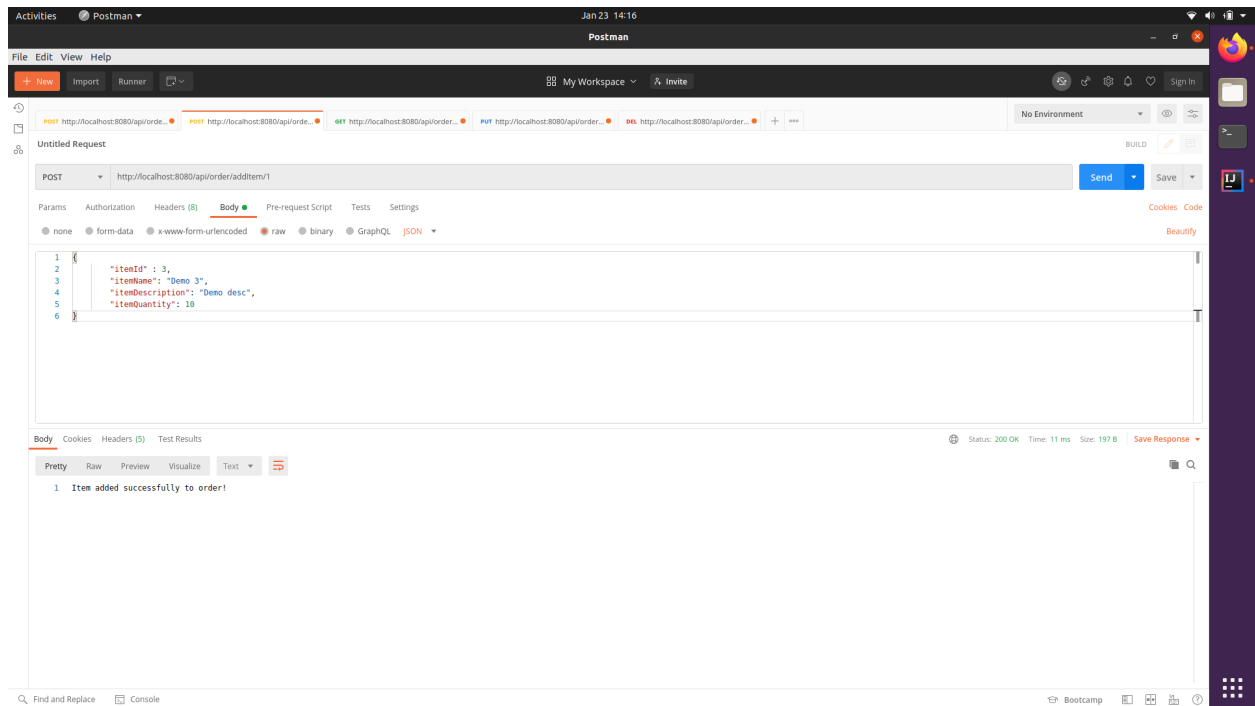
<http://localhost:8080/api/order/updateItemQuantity/{orderId}/{itemId}/{newQuantity}>

### Screenshots from Postman:

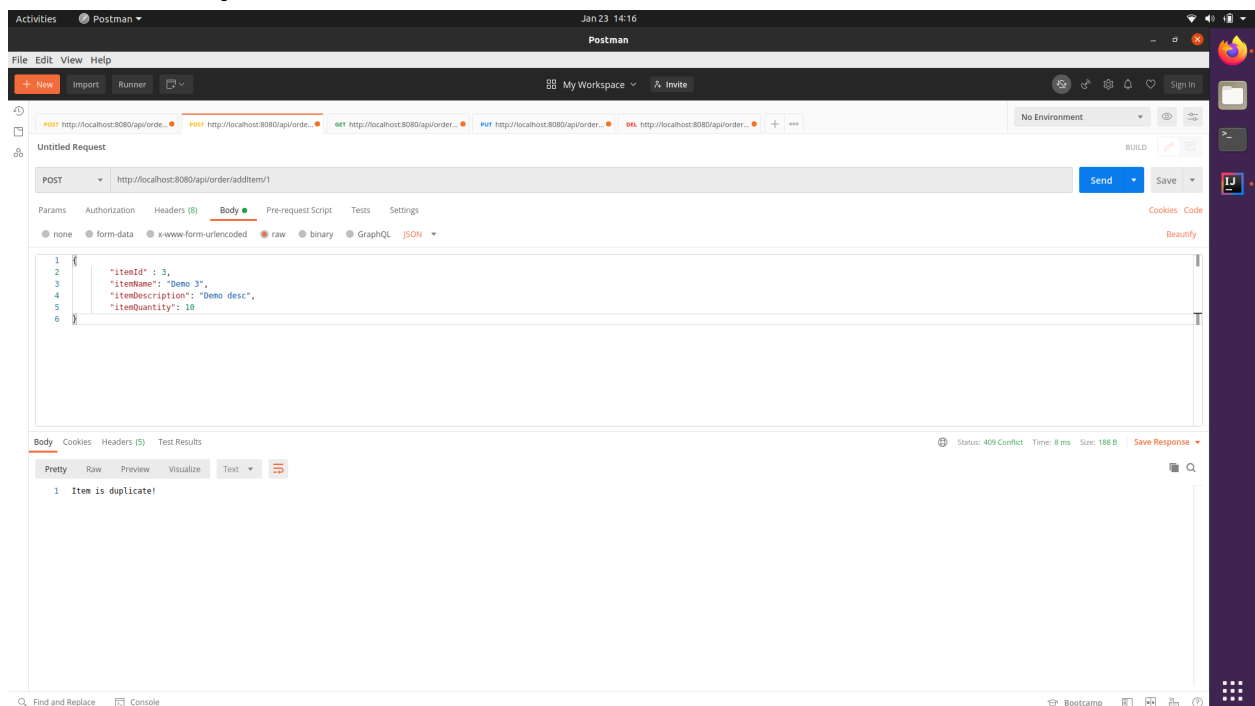
#### 1. Create a new order



## 2. Add item



## 3. Add duplicate item



## 4. DeleteItem

The screenshot shows the Postman application interface. At the top, the title bar indicates 'Jan 23 14:17' and 'Postman'. The main window has a menu bar (File, Edit, View, Help) and a toolbar with buttons for '+ New', 'Import', 'Runner', and 'My Workspace'. Below the toolbar, there's a tab for 'Untitled Request'. The request method is set to 'DELETE' and the URL is 'http://localhost:8080/api/order/deleteItem/1/2'. The 'Params' tab is active, showing a table with columns 'KEY', 'VALUE', and 'DESCRIPTION'. The 'Body' tab is also visible, showing the response '1 Item removed successfully!'. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 18 ms', and 'Size: 190 B'.

KEY	VALUE	DESCRIPTION
Key	Value	Description

1 Item removed successfully!

## 5. Get Order

The screenshot shows the Postman application interface. At the top, the title bar indicates 'Jan 23 14:16' and 'Postman'. The main window has a menu bar (File, Edit, View, Help) and a toolbar with buttons for '+ New', 'Import', 'Runner', and 'My Workspace'. Below the toolbar, there's a tab for 'Untitled Request'. The request method is set to 'GET' and the URL is 'http://localhost:8080/api/order/getOrder/1'. The 'Body' tab is active, showing the response in JSON format. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 7 ms', and 'Size: 350 B'.

```
1 {
2   "orderId": 1,
3   "itemList": [
4     {
5       "itemId": 1,
6       "itemName": "Demo",
7       "itemDescription": "Demo desc",
8       "itemQuantity": 10
9     },
10    {
11      "itemId": 3,
12      "itemName": "Demo 3",
13      "itemDescription": "Demo desc",
14      "itemQuantity": 10
15    }
16  ]
17 }
```

## 6. Order not found

The screenshot shows the Postman application interface. At the top, the title bar indicates 'Postman' and the date 'Jan 23 14:16'. The main workspace displays an 'Untitled Request' with a GET method and the URL 'http://localhost:8080/api/order/getOrder/2'. The 'Body' tab is selected, showing a message: 'This request does not have a body'. Below the request details, the 'Test Results' section shows a status of '404 Not Found' with a time of '6 ms' and a size of '187 B'. The response body is displayed in the 'Pretty' view, showing the text '1 Order not found!'. The bottom status bar includes a search icon, 'Find and Replace', 'Console', and a 'Bootcamp' logo.

## 7. Order duplicate

The screenshot shows the Postman application interface. At the top, the title bar indicates 'Postman' and the date 'Jan 23 14:16'. The main workspace displays an 'Untitled Request' with a POST method and the URL 'http://localhost:8080/api/order/create'. The 'Body' tab is selected, showing a JSON payload in the 'Pretty' view:

```
1 {
2   "orderId": 1,
3   "itemList": [
4     {
5       "itemId": 1,
6       "itemName": "Demo",
7       "itemDescription": "Demo desc",
8       "itemQuantity": 10
9     },
10    {
11      "itemId": 2,
12      "itemName": "Demo 2",
13      "itemDescription": "Demo desc ",
14      "itemQuantity": 1
15    }
16  ]
17 }
```

Below the request details, the 'Test Results' section shows a status of '409 Conflict' with a time of '6 ms' and a size of '189 B'. The response body is displayed in the 'Pretty' view, showing the text '1 Order is duplicate!'. The bottom status bar includes a search icon, 'Find and Replace', 'Console', and a 'Bootcamp' logo.

## 8. Update item quantity

The screenshot shows the Postman application interface. At the top, the title bar indicates 'Activities', 'Postman', and the date 'Jan 23 14:17'. The main menu includes 'File', 'Edit', 'View', and 'Help'. Below the menu, there's a toolbar with buttons for '+ New', 'Import', 'Runner', and 'My Workspace'. The 'My Workspace' dropdown is currently open, showing a list of requests with their methods and URLs. The selected request is a PUT request to 'http://localhost:8080/api/order/1/3/100'. The 'Send' button is visible next to the URL.

The 'Untitled Request' section is active, showing the 'PUT' method and the URL 'http://localhost:8080/api/order/updateItemQuantity/1/3/100'. The 'Params' tab is selected, showing a table for 'Query Params'.

KEY	VALUE	DESCRIPTION
Key	Value	Description

The 'Body' tab is also visible, showing a 'Pretty' view of the response. The response is a JSON object: 

```
{  "1": "Item quantity updated successfully!"}
```

. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 10 ms', and 'Size: 199 B'. There are also buttons for 'Save Response', 'Find and Replace', and 'Console'.